(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
1 March 2001 (01.03.2001)

PCT

(10) International Publication Number
WO 01/15449 A1

(51) International Patent Classification[7]: H04N 7/173, 7/16

(21) International Application Number: PCT/IB00/01131

(22) International Filing Date: 17 August 2000 (17.08.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/149,698 20 August 1999 (20.08.1999) US
09/602,342 24 June 2000 (24.06.2000) US

(71) Applicant *(for all designated States except US)*: SINGULARIS S.A. [CH/CH]; 8, côtes de Montbenon, CH-1003 Lausanne (CH).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: VAMPARYS, Franck [CH/CH]; Chemin du Croset 8, CH-1024 Ecublens (CH).

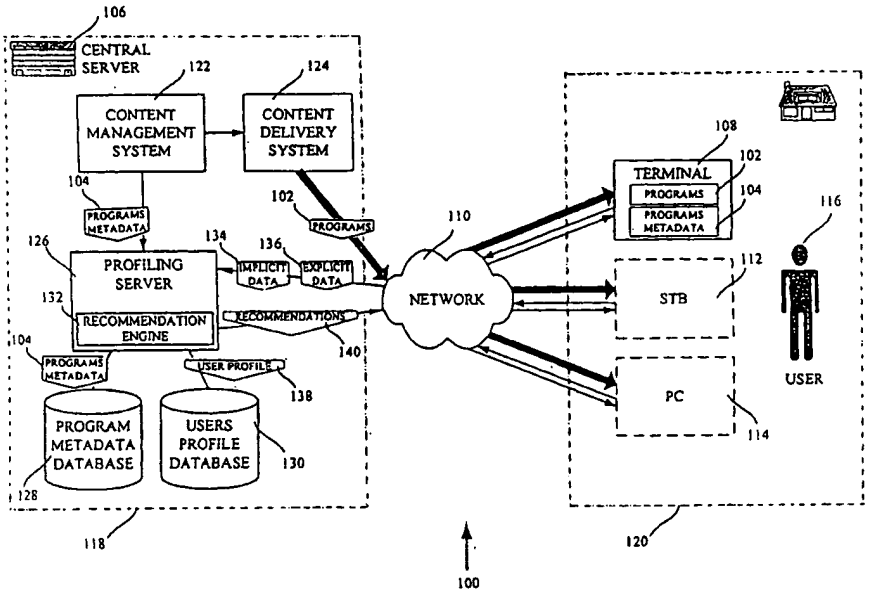
(74) Agent: MOETTELI, John; Bugnion S.A., Case Postale 375, CH-1211 Genève 12 (CH).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE,

[Continued on next page]

(54) Title: METHOD AND APPARATUS FOR CREATING RECOMMENDATIONS FROM USERS PROFILE BUILT INTERACTIVELY



(57) Abstract: A system is provided for generating program recommendations for a plurality of users based on program metadata and user profiles. The system uses a central prolifing server unit that includes a program agent system, a user agent system, and a recommendation engine. The program agent system has means for importing program metadata from a content management system and for storing the program metadata into a program metadata database. The user agent system includes (i) means for collecting explicit and/or implicit information on program metadata and users; (ii) means for storing the explicit and/or implicit information into a user profile database; and (iii) means for transmitting a list of program recommendations for users to user terminals. The recommendation engine includes a (i) means for building program recommendations for users based on program metadata and user profiles, through use of a plurality of filtering engines for evaluating user profiles, with a schedule of available programs; (ii) means for gathering and weighting the built program recommendations in order to generate a list of program recommendations per user; and (iii) means for storing said list of program recommendations in a user profile database.

BEST AVAILABLE COPY

WO 01/15449 A1

IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:

— *With international search report.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

METHOD AND APPARATUS FOR CREATING RECOMMENDATIONS FROM USERS PROFILE BUILT INTERACTIVELY

Background of the Invention

This invention relates to entertainment media management systems, and, more particularly, to an apparatus and method that recommends a program to a user based on program metadata and on user profiles.

In order to better understand this disclosure, the following definitions are offered. A program is a media (most likely an electronic media such as music, television programs, digital video or a tangible item such as a book, compact disc or a live event such as a play, opera, etc.) with validity periods. Validity periods define time periods when the program can be delivered to users. Program metadata are data that describes a program.

Recently, breakthroughs in digital technologies allow users to receive hundreds of video channels. In fact, it is widely believed that the barrier of 1000 channels will soon be broken. In this new environment, the amount of content available is overwhelming. Electronic guides are limited by the amount of information that can be displayed on a single screen at a given time. Additional tools are therefore required in order to help the user find a program that he is interested in. One way to do this is to propose recommendations on programs addressing an individual user's preferences (for example user tastes, behavior, demographic information). The prior art includes various systems directed toward enhancing television-viewing efficiency by providing individual suggestions. For example, an early system described in US Pat. No. 4,170,782, the content of which is incorporated herein by

reference, allows the viewer to preselect a television-viewing schedule of desired television channels to be viewed during selected time periods. This system also monitors the television viewer and relays this information to a central data processing center over a communications link. Other similar systems, such as those described in US Pats no. 4,264,924 and 4,706,121, the contents of which are incorporated by reference, facilitate access to programming but provide no assistance to the user in determining what programming to select for subsequent viewing. US Pat. No. 4,745,549 and 5,075,771, the contents of which are incorporated herein by reference, try to provide selective access to programming from user preferences built on a user questionnaire. However, this kind of interaction based on a general questionnaire does not provide any insights as to the desirability of a particular program.

The system described in US Pat. No. 5,223,924, the content of which is incorporated by reference, begins to address the above problem by providing a user interface for automatically correlating the user preferences with TV program information and creating and displaying a personalized user program database based on the results of the correlation using a "free-text search" engine. In this system, the user profile is limited to automatically update the database of "preferred" programs using direct user feedback.

US Pat. No. 5,410,344, the content of which is herein incorporated by reference, provides a system for selecting programs for presentation to a viewer. The programs selected from a viewer preference file built with explicit ratings on program metadata and from a neural network engine. A neural network is an black box of processing elements with a limited number

of inputs and outputs. These processing elements are able to "learn" by receiving weighted inputs that, with time and repetition, can be made to produce appropriate outputs. The viewer preference file is accessible only locally on the viewer terminal and contains no implicit ratings so that the value and efficiency of recommendations are limited.

US Pat. No. 5,534,911, the content of which is incorporated by reference, describes a virtual personal channel but user profile is stored locally and is non persistent between sessions. In U.S. Pat. No. 5,801,747, the content of which is incorporated by reference, a user-profiling server is described. However, user profiles are built only based on viewing activities of the user. Further, the system can only display preferred channels or categories.

Some prior art systems address the problem of suggesting programs from a large program database such as a movie database (for video on demand) as described in U.S. Pat. No. 5,483,278, the content of which is incorporated by reference. In this system, the user makes explicit ratings on program metadata. The system presents programs in an individual order and the user can rate them explicitly (e.g., by selecting "liked" or "disliked") and the order of program presentation to the user is modified. The user profile is stored locally with respect to the terminal and is non persistent so that the interest of such a system is limited.

Referring to US Pat. No. 5,758,257 and US Pat. No 6,020,883(the content of which is incorporated by reference), two profiles are built from explicit ratings, one on content data and one on viewer's preferences. An "agreement matrix" is then calculated by comparing the two profiles and a

"virtual channel" is generated, providing personal recommendations on programs to each viewer. A feedback path with monitoring information is also provided so that user profiles and content profiles can be improved. This system works also with movies, music or book databases (kiosk). This system includes an "agreement matrix" and is well adapted to work inside a powerful terminal but not on a profiling server with millions of users and a limited resource terminal.

In US Pat. No. 5,798,785, the content of which is incorporated herein by reference, a terminal offers the ability to assist a viewer in choosing programs for viewing. Program recommendations are built from information on frequently viewed programs, persistent profile information and user mood information. This information is interpreted into preferred program indicators and matched with the program metadata to suggest a list of programs. The first limitation of this system is that it is not able to make suggestions by correlated profiles of different users. The second limitation is that the data gathering and analysis techniques used are adequate for a powerful terminal but not for a central profiling server with millions of users.

Therefore, what is needed is a system and method which overcomes the drawbacks of the prior art, generating user program recommendations from the knowledge of the user profile itself, and also from the profiles of its neighbors that have the same tastes.

Summary of the Invention

A computer-readable medium is encoded with a method of profiling that collects explicit and implicit data on users related to programs that can be accessed by users and to provide program recommendations to each user.

The system and method is designed for the server-side and limited resource terminals to set up user profiles from explicit and implicit data simultaneously on a full program schedule database as well as a specific program (e.g., movie) database.

Schedules on program availability and program metadata are broadcast to or received on request by the user terminal through a communication medium from a central system that contains a network of physical servers (network, application, database server for example). When, from a terminal, a user interactively rates a program or a field of the program metadata, this explicit rating is sent to the profiling server through the return path and is stored in the user's profile database. The profiling server collects information on user tastes with respect to program metadata. The profiling server also receives implicit information from user behavior such as user program remind actions, user purchase actions, or user monitoring information. All these implicit ratings complete the profile of each user. Then the recommendation engine of the profiling server evaluates the profile of users with the schedule of programs that are now available and that will be available in the near future. Based on the evaluation process, the recommendation engine builds recommendations of programs for users. The recommendation engine works with several different filtering engines such as a content filtering engine and a collaborative filtering engine and treats

one user at a time in each filtering engine. Recommendations are weighted and gathered at the output of these filtering engines and a unique recommendation per program is generated by the recommendation engine for each user. Then the list of program recommendations is stored in the profile of the user. These recommendations are sent to each user terminal on terminal request or by a push from the profiling server. The user can access the resulting recommendations through a user interface that displays recommendations of the best programs for him with different scenarios such as portal, menus, tickers, virtual channels, and lists.

An object of the invention is to provide a system and method for server-side and limited resource terminals to set up user profiles from explicit and implicit data simultaneously on a full program schedule database as well as a specific program database. Another object of the invention is to generate user program recommendations from the knowledge of the user profile itself, and also from the profiles of its neighbors that have the same tastes.

Brief Description of the Drawings

The features and advantages of the present invention will become more apparent and understood from the following detailed description which should be read in conjunction with the accompanying drawings, in which:

FIG. **1** is a block diagram of a profiling and recommendation system;

FIG. **2** is a schematic diagram of a terminal screen of the invention;

FIG. **3** is a flow chart of the method of collecting the explicit ratings of the user on program metadata;

FIG. **4** is a flow chart of the method of collecting implicit data on the programs the user has watched by storing the switches between channels;

FIG. **5** is a flow chart of the method of collecting implicit data on the programs the user has watched by evaluating the user's viewing time;

FIG. **6** is a block diagram of a system of the invention including a two-way profile server that receives, stores program metadata and user explicit/implicit data and then builds and returns program recommendations;

FIG. **7** is a block diagram of a recommendation engine that provides program recommendations from a set of filtering engines;

FIG. **8** is a flow diagram of a method for selecting programs from the user profile and from program metadata in the content filtering engine;

FIG. **9** is a block diagram of a user terminal including network connections and user interactions;

FIG. **10** is a representative program description guide with explicit ratings displayed on top of a TV program.

Detailed Description of the Preferred Embodiment

The following embodiments of the present invention will now be described in detail relative to Figures 1 to 10.

Referring now to Fig. 1, the present invention **100** is shown as part of an expanded program delivery system **100** that delivers programs **102** and program metadata **104** from a central server **106** to a user terminal **108** through a network **110**. In the case of a cable television system, the central server **106** can be located at the operator head-end and the user terminal **108** can be an operator set-top box

(STB) or the television itself. In the case of an Internet solution, the central server **106** can be integrated into the web and application servers and the terminal **108** can be a personal computer (PC) **114**. The terminal **108** is a device that displays programs **102** and program metadata **104** to a user and that can interact with the user. In the television field, the terminal **108** is usually a set-top box (STB) or the television set. The terminal **108** can also be a wireless device (PDAs, mobile phones, etc.).

Major System Components

Referring again to Fig. **1** , the system **100** operates within the context of a content or service provider system for communicating programs **102** and programs metadata **104** to users **116**. This system **100** is composed of two parts **118** and **120**. The first part **118** is the central server **106** that manages and delivers content to the users **116**. The second part **120** is the user terminal **108** that receives the content and allows the user **116** to interact with the central server **106**. A content management system **122** manages programs **102** and programs metadata **104** and is synchronized with the content delivery system **124** that sends programs **102** to the network **110**, destined to the users **116**. The content management system **122** also sends program metadata **104** to the profiling server **126**. The profiling server **126** collects information on programs metadata **104** and on user profiles and stores these data in the program metadata database **128** and in the users profile database **130**. The profiling server **126** is the main component of the invention: from the collected data, the server **126** builds and delivers program recommendations to each user **116**. The recommendations are received by the terminal **108**. A user **116** can use different terminals **108** to access its profile on the profiling server **126**. The profiling server **126** imposes no constraints

on the infrastructure of the network **110** if we assume a reasonable bandwidth adequate for the number of users **116**.

It should be noted that program metadata **104** may also be obtained from other sources, such as a ftp xml file or in the media itself, digitally embedded in the content and need not come from the content management system **122**.

Program Metadata and User Profiles

A program **102** is described by its metadata **104**. Metadata **104** of programs contain a large set of possible fields but only a few subsets of this fields are useful for the profiling server **126** and its recommendation engine **132**. A more detailed description of a preferred embodiment is provided as follows.

(a) Program ID **218**: The Program ID **218** uniquely identifies a particular program **102**. The ID **218** can be an integer (32 bits) or a long integer (64 bits) or a string of bytes not exceeding 2000 bytes.

(b) Title **208**: The title **208** uniquely identifies the main title of the program **102** and is defined as a string.

(c) Category **210**: The category **210** identifies the media content and media type category of a program **102**. The media type category defines the type of program **102** such as a movie, song, a documentary, a live event, a software, etc. The media type category is defined in one level, which means it is one element of a list of category values. The media content category indicates the nature of the content of the program **102**. It consists of three

hierarchical level categories. For example, a program **102** might have the media content level categories of (1) sport, (2) football, and (3) Arsenal. In the case of an Electronic Program Guide (EPG), the two higher-level fields can be compatible with category fields defined in the Digital Video Broadcasting Standard (DVB). Category fields are defined as strings. Optionally, a unique ID can be added to the string definition.

(d) Description **220**: The description **220** describes the program **102** in a textual form and is defined as a string. This field is optional for the recommendation engine but useful for the users **116**.

(e) People **212**: People **212** identify the stars or other persons of interest in the program **102**. One individual is defined by a people category and by name. Examples of people category are: actor, producer, presenter, famous, etc. People categories and people names are designed as strings. Optionally, a unique ID can be added to the string definition.

(f) Keywords **214**: Keywords **214** describe the program **102** by a list of keywords. Each keyword represents the content of one part of the program **102**. These keywords can be interpreted without the category fields. For example, a keyword might be the "Louvre" in the context of the category "documentary, travel, France". Optionally, a unique ID can be added to the string definition of the keyword.

(g) Parental rating **216**: Parental rating **216** identifies restrictions regarding who can access the associated program **102**. For example, the rating might be "users older than 13". Classification can be by age or based upon content such as violence, language, nudity and/or sex. Parental control

attributes are defined as strings. Optionally, a unique ID can be added to the string definition.

User profiles are built on explicit and implicit data **134**, **136** collected by the profiling server **126**. Explicit data **136** is data that are produced by a direct interaction with the user **116**. Each user **116** can indicate using a rating scale his preferences on or his reaction to a specific program **102** or a program category or people or keywords. All the user ratings using the rating scale are collected by the profiling server **126** and are stored in the users profile database **130**. When a user **116** purchases a program **102** or makes a remind action on a program, these actions are also stored in the user's profile database **130**. Implicit data **134** is data that is monitored by the terminal **108** and sent to the profiling server **126**. This data **134** is also stored in the user's profile database **130**. Implicit data **134** gives information on the viewing habits of the user **116**, such as which programs **102** he has watched or used, and the time of day the program was viewed.

The user profile database **130** is structured as follows:

(a) Basic data: When users buy or log on to the terminal **108**, he must identify himself. The profiling server **126** collects this information. Examples of this information are: nickname, age, zip code, gender, income, etc.

(b) History data: This data contains all the explicit and implicit data **134** and **136** collected.

(c) Processed data: This data contains information that is generated by the profiling server **126**. For example, this includes program recommendations data generated for each user.

(d) Aggregated data: This data contains data that are extracted and aggregated based on a user profile **138**. In this aggregated data, reference to an individual user **116** is lost.

Referring now to FIGs. **2** and **3**, a submethod **300** is described that acquires explicit data **136** by collecting the user's tastes by means of a rating process. In a first step **302**, the terminal **108** (shown in FIG **2**) receives program metadata **104** and stores the data locally. In a second step **304**, the metadata is displayed when necessary or requested. In a third step **306**, following the display of the metadata **104**, the user can select a field **214** of a displayed program **200**, assign a grade **206** for that field **214** and validate this rating using a validate button such as that on a remote control or touch screen. In a fourth step **308**, the rating **206** is saved locally on the terminal **108**. In a fifth step **310**, the rating is sent to the profiling server **126** asynchronously. A delay can be added before the transmission of the profiling server **126** in order to facilitate the management of the load of the server. When the delay is set to zero, optionally, the profiling server **126** can activate the recommendation engine **132** and immediately returns program recommendations **204** generated from the new rating. Fields **222** that can be rated are the metadata **104** available for each program **200** such as title, categories, people, keywords, etc. The same rating process also applies to purchases and reminder requests acquisition.

Referring now to FIG. **4** , a submethod **400** is described for acquisition of implicit data **134** on a terminal **108**. In a first step **402**, a time counter is initialized and the system is waiting a user action. In a second step **404**, the user changes a channel. In a third step **406**, an evaluation of the

manner in which channels are changed (the "channel zapping") is performed. This evaluation determines the time spent on the previous channel. In a fourth step **408**, the duration between zapping is evaluated. In a fifth step **410**, if the duration between zapping is greater than the limit, zapping information is accepted and is saved locally **120**. In a sixth step **412**, the method collects a predefined (configurable) number of channel changes/zappings on the terminal **108**. In a seventh step **414**, the method sends this information to the profiling server **126** asynchronously. To use implicit data **134**, the recommendation engine **132** does not only need to know zapping information but also how long a user has watched a program **200**. This information can be built from the zapping information and from the program schedule on the server side **106**. However, this additional information can also be built up in the terminal **108**.

Referring now to FIG. **5**, this last option has the advantage of reducing the load of the server **106**. The initial steps **502**, **504**, **506** of this submethod **500** are the same as that of FIG. **4** and this submethod works in parallel with the main method shown in FIG. **4**. After the channel zapping evaluation **506**, an evaluation step **508** is added which evaluates the programs **200** watched. This evaluation step **508** determines how long the current program **200** of the previous channel has been watched by the user. In a time check step **510**, if the program **200** is terminated, the time the program was watched is evaluated to determine whether it exceeds the predetermined limit (which is configurable). If so, in a saving step **512**, the program-watched information is stored locally on the terminal **108**. In a collection step **514**, the method returns to the first step **502** until the number of program **200** watched reaches a maximum. In a subsequent step **516**, after having collected a predefined

(configurable) number of watched program information on the terminal **108**, the information is sent to the profiling server **126** asynchronously.

Profiling System

Referring now to FIG. **6**, a block diagram of the preferred embodiment of the profiling server **126** is described. The profiling server **126** includes a core platform system **602** that synchronizes all the subsystems (**132**,**604**,**606**) of the profiling server **126**. First, a program agent system **604** receives program metadata **104** from the content management system **122**, parses the program metadata **104**, converts the program metadata into an internal format as described earlier, and stores the new program metadata in the program metadata database **128**. Only the program agent system **604** has to be adapted when a different content management system **122** is connected to the profiling server **126**. In a matter similar to the program agent system **604**, a user agent system **606** is the gateway between the user terminal **120** and the users profile database **130**. The user agent system **606** receives explicit and implicit data **136** and **134** respectively from the user terminal **120**. It identifies, validates and then stores the data in the user profile database **130**. If the terminal **120** requests immediate program recommendations built on the newly received data, the user agent system **606** sends the request to the core platform system **602** that activates the recommendation engine **132**. The recommendation engine **132** returns the program recommendation list to the core platform system **602** that sends the recommendations to the terminal **120** through the user agent system **606**. The core platform system **606** controls and manages program and user agent sys-

tems **604** and **606**, respectively, as well as the recommendation engine **132**. The profiling system **126** has been designed as a flexible architecture.

Recommendations Engine

Referring now to FIG. 7, a block diagram of the recommendation engine **132** is shown. The engine **132** has two working modes. In a first working mode, called a low priority batch mode, the recommendation engine **132** builds new program recommendations for each user when the core platform system **602** informs it that the new program metadata **104** has been added to the program metadata database **128**. Depending on the size of the user's profile database **130**, the update of each user can take a long time. A second mode with a high priority allows the profiling system **600** to respond faster to a request of a user. In this mode, the profiling server **126** brings instant gratification of the user by giving him program recommendations related to the new rating he has sent to the profiling server **126**. By evaluating one user at a time, the recommendation engine **132** can easily be adapted to multi-processors and multi-servers architecture with hash coding techniques on users.

In a first substep **702**, the recommendation engine **132** selects the user it wishes to treat. In a second substep **704**, once the user is selected, the recommendation engine **132** reads the user profile stored in the users profile database **130**. Then the recommendation engine **132** activates different filtering engines **706**, **708** and **710** with that profile. The recommendation engine **132** supports multiple algorithms and can aggregate the result of the

different filtering engines **706**, **708** and **710** to a single list of program recommendations. In a third substep **720**, the user profile in the user profile database **130** is updated. In order to accomplish this, the weighting coefficients **712**, **714** and **716** are added to each respective program recommendation of each filtering engine **706**, **708** and **710**. Coefficients **712**, **714** and **716** differ depending on the filtering engine **706**, **708** or **710** to which they apply and also depending on the program content category because a filtering engine can be better adapted to one content category than another content category. For example, the collaborative filtering engine **708** gives better results with the movie content category than with the sport content category.

The coefficients **712**, **714** and **716** can be tuned manually. They can also evolve dynamically by using a feedback adaptive algorithm well adapted to heuristic algorithms. The recommendation engine **132** adapts the coefficients **712**, **714**, and **716** by using for all users either an evaluation of the difference between the recommendation grade and the grade the user has given to that program or an evaluation of the difference between the recommendation grade and the duration the program **200** has been watched by the user. By default, the recommendation engine **132** uses two complementary filtering algorithms: content filtering and collaborative filtering. The content filtering engine **706** generates recommendations separately for each user while the collaborative filtering engine **708** finds correlation between the user tastes. The collaborative filtering requires bootstrap to start to generate recommendations while the content filtering does not. The content of the filtering engine **706** can generate program recommendations as soon as at least one rating has been made by the user

116. Of course, if only a few ratings are given, the customization of the programming recommendations is low.

One solution consists of providing a default profile when the user profile is first created. This default profile can be built on the basic data mentioned under the heading "Program Metadata and User Profiles", available on the user **116** and on aggregated data such as the best or the most rated programs by other users. Another more interesting solution is to leave the choice to the user **116** to choose, as a default profile, the profile of a famous person he likes. Collaborative filtering has proven useful for movie, book, music or documentaries. Content filtering has powerful availability with news, sport, people, and all strongly categorized items. In the TV context it also fits for periodic programs **102**. The collaborative filtering engine **708** requires more intensive CPU resources than the optimized content filtering engine **706**, the collaborative filtering engine **708** can only be used for a subset of all the programs **102** (e.g. movies, documentaries, music). These two algorithms respectively have proved to be well adapted to item recommendation and are complementary. Nevertheless, the recommendation engine **132** can easily integrate another filtering engine such as a more general social filtering engine **710**. Each filtering engine **706**, **708**, **710** has access to the program metadata database **128** with the explicit and implicit data **136** and **134**.

FIG. **8** describes the internal architecture of the content filtering engine **706**, custom developed according to the needs of the present invention. The engine **706** is optimized for structured and categorized metadata. When the content filtering engine **706** is started, program metadata

104 with a right validity period are selected in a selection step **810** and indexed in an indexing step **812** for each program field such as "title", "people", "content category", and "keyword" used by the engine. From the user profile already loaded **704**, the engine **706** separates information on user tastes **802**, user purchases **804**, user reminder requests **806** and use monitoring information **808**. There is no technical limitation to the number of subjects for possible characterization or categorization -other user information can be used by the engine **706**. Each information category **702**, **704**, **706** and **708** goes through a tuned matching engine for tastes **814** for purchases **816** for reminder requests **818** and for monitoring **820**. Each matching engine **814**, **816**, **818** and **820** uses the same algorithm based on static matching rules and non-linear distance estimation between program metadata **104** and user profile fields (defined in user profiles section) but with different rules and distance estimation coefficients for each matching engine. Rules and distance estimation coefficients are based on heuristics built on the knowledge acquired through development, study, and experimentation. Use of matching rules allows the purveyor to find a balance between search (Indexes) and knowledge (Rules), the two main approaches of AI. Details on rule-based algorithm can be found in Charles Forgy, "The Rete Algorithm: A Fast Algorithm for the Many Pattern / Many Object Pattern Matching Problem", 1979, PhD thesis, Carnegie-Mellon University, the content of which is incorporated herein by reference. Due to the indexing techniques (read access in O(1)) used on program metadata **104**, the content filtering engine **706** is very fast and can support a large number of users **116**. The bottleneck is on database access and not on the engine **706**. Structured program metadata **104** is key to guaranteeing the satisfaction of program

recommendations **204** and to guaranteeing a fast content filtering engine **706**. Programs suggested by the matching engines **814**, **816**, **818** and **820** are then weighted using a weight coefficient **822**, **824**, **826** and **828** respectively and aggregated in a list of program recommendations. A first level **830** aggregates program recommendations built from the user behavior (purchase, remind, monitoring) and a second level **832** aggregates these recommendations with the recommendations built in the taste matching engine **814**. The matching engines (**814**, **816**, **818** and **820**) take into account at what time the program **200** has been watched, reminded and/or purchased. For example, some users **116** watch movies in the evening and news in the morning while others watch news in the evening. The recommendation engine **132** tries to follow this time pattern when it generates the program recommendations. The content filtering engine **706** generates as output a program recommendation list **906** with a grading **920** on the programs of the list **918**.

User Terminal

Now referring to FIG. **9**, the data flow of the terminal **120** is shown. The terminal **120** is a device that displays programs **102** and program metadata **104** to a user **116**, collects and sends explicit and implicit data **136** and **134**, respectively, on users to the profiling server **126**. A user **116** interacts with the profiling server **126** through the terminal **120** that displays its personal recommendations **204**.

Now referring to FIG. **10**, a real case of the process of interaction with the user **116** is shown. The terminal **120** displays a program description panel

902 with information on one program on top of another. Metadata **104** is displayed, including the program title **908**, the content category **910** and a list of people **912**. The rating panel **904** applied on the selected item **914** ("Peter Barton" in this example) and the user **116** can change the grade of the explicit rating **916**. Program recommendations available for the current user **116** are displayed in the recommendation panel **906**. Each program recommendation is represented by its title **918** and its grade **920**, in this case, one to five stars. Different processes of interaction can also be used. For example, the program recommendations **906** can be displayed in a portal, on menus or be received by e-mail.

Multiple variations and modifications are possible in the embodiments of the invention described here. Although certain illustrative embodiments of the invention have been shown and described here, a wide range of modifications, changes, and substitutions is contemplated in the foregoing disclosure. In some instances, some features of the present invention may be employed without a corresponding use of the other features. Accordingly, it is appropriate that the foregoing description be construed broadly and understood as being given by way of illustration and example only, the spirit and scope of the invention being limited only by the appended claims.

What is claimed is:

1. A system for generating program recommendations for a plurality of users based on program metadata and user profiles, the system comprising a central profiling server unit, the unit comprising:

(a) means for importing program metadata from a program metadata source;

(b) means for storing the program metadata into a program metadata database;

(c) means for collecting explicit and/or implicit information on program metadata from user interaction therewith;

(d) means for storing the explicit and/or implicit information into a user profile database;

(e) means for building program recommendations for users based on program metadata and user profiles, through use of a plurality of filtering engines for evaluating user profiles with a schedule of available programs;

(f) means for gathering and weighting the program recommendations in order to generate a list of program recommendations per user; and

(g) means for transmitting a list of program recommendations for users to user terminals for display.

2. The system of claim 1, further comprising means for storing said list of program recommendations in a user profile database.

3. The system of claim 1, wherein the means for importing program metadata from a content management system and the means for storing the program metadata into a program metadata database are comprised in a program agent system.

4. The system of claim 1, wherein the means for collecting explicit and/or implicit information on program metadata and users, the means for storing the explicit and/or implicit information into a user profile database, and the means for transmitting a list of program recommendations for users to user terminals are comprised in a user agent system.

5. The system of claim 1, wherein the means for building program recommendations, the means for gathering and weighting the built program recommendations, and the means for storing said list of program recommendations in a user profile database are comprised in a recommendation engine.

6. The system of claim 1, wherein the program metadata source is a content management system.

7. The system of claim 1, further comprising a terminal device having:

(a) means for receiving program metadata and user program recommendations;

(b) means for displaying the program metadata and user program recommendations; and

(c) means for collecting and sending explicit and/or implicit information on program metadata and users to the central profiling server.

8. The system of claim 7, further comprising means for storing the program metadata and user program recommendations for later access.

9. A computerized method encoded on a computerized medium, the method providing program recommendations to a user based on program metadata and user profile and comprising the steps of :

(a) accessing a server-side program metadata database;

(b) presenting the program metadata on a user terminal for user selection and rating;

(c) collecting explicit and/or implicit information on program metadata and the user;

(d) creating at least one user profile based on collected user explicit and/or implicit information;

(e) comparing the program metadata with the user profile for building program recommendations using (i) a recommendation engine supporting a plurality of filtering engines at the same time, generating a program recommendation list associated with each filtering engine and the user; (ii) a gathering and weighting engine, applied to the filtering engines output, for generating a list of program recommendations per user;

(f) transmitting the list of program recommendations to the user terminal; and

(g) presenting the list of program recommendations to the user.

10. The method of claim 9, further comprising the step of storing on the server side the list of program recommendations.

11. The method of claim 9, further comprising the step of :

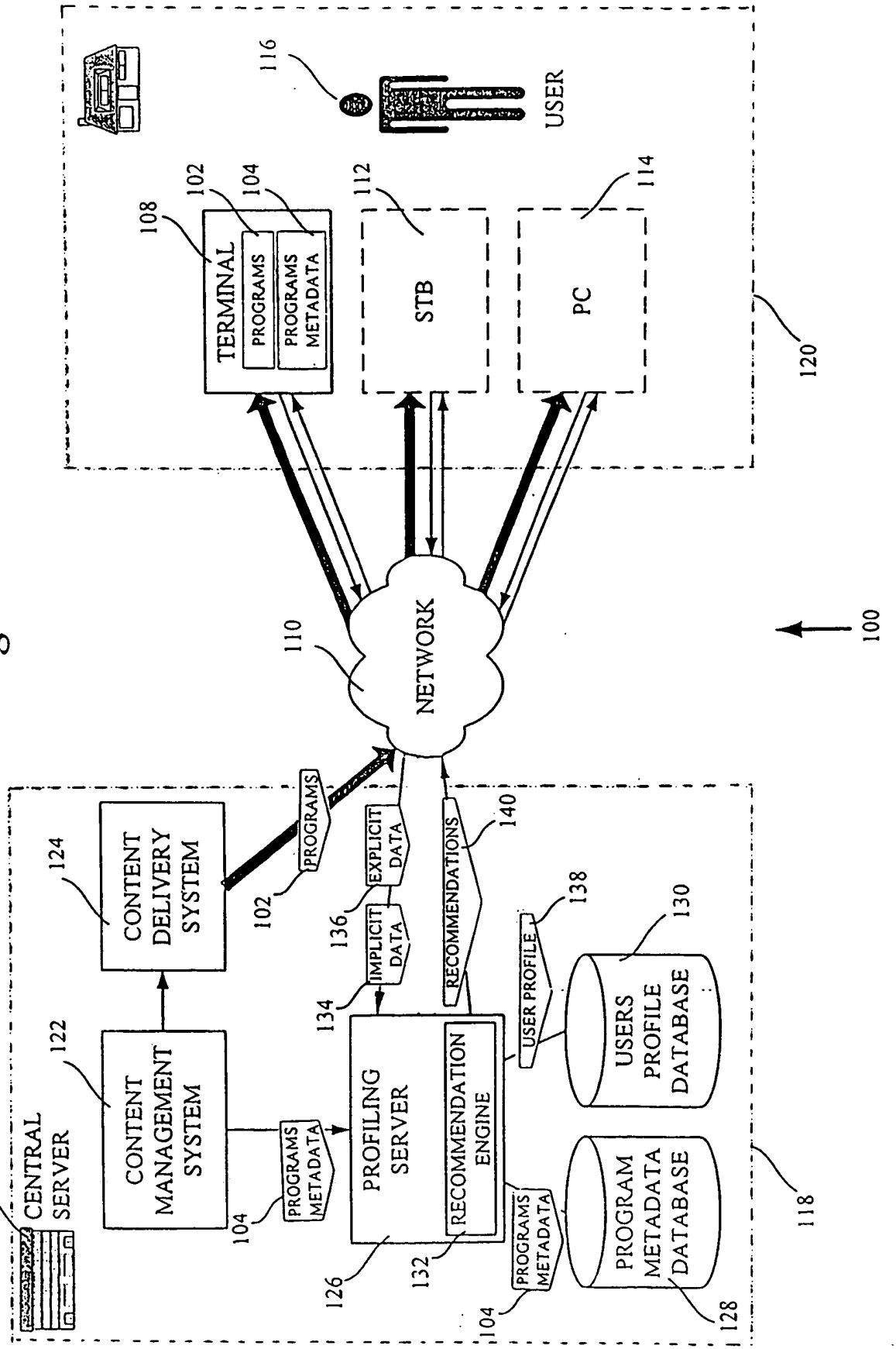
(a) receiving on terminal-side program metadata and user program recommendations;

(b) displaying on terminal-side the program metadata and user program recommendations; and

(c) collecting and sending from terminal explicit and/or implicit information on program metadata and users to the central profiling server.

12. The method of claim 9, wherein filtering engines include a content filtering engine using indexing techniques on program metadata and matching engines based on static matching rules and non-linear distance estimation between program metadata and the user profile.

Fig. 1


116
USER
102
104
108
TERMINAL
PROGRAMS
PROGRAMS METADATA
112
STB
114
PC
120
NETWORK
110
100
CENTRAL SERVER
106
CONTENT MANAGEMENT SYSTEM
122
CONTENT DELIVERY SYSTEM
124
PROGRAMS
102
EXPLICIT DATA
136
IMPLICIT DATA
134
RECOMMENDATIONS
140
USER PROFILE
138
USERS PROFILE DATABASE
130
PROGRAMS METADATA
104
PROFILING SERVER
126
RECOMMENDATION ENGINE
132
PROGRAM METADATA DATABASE
128
118

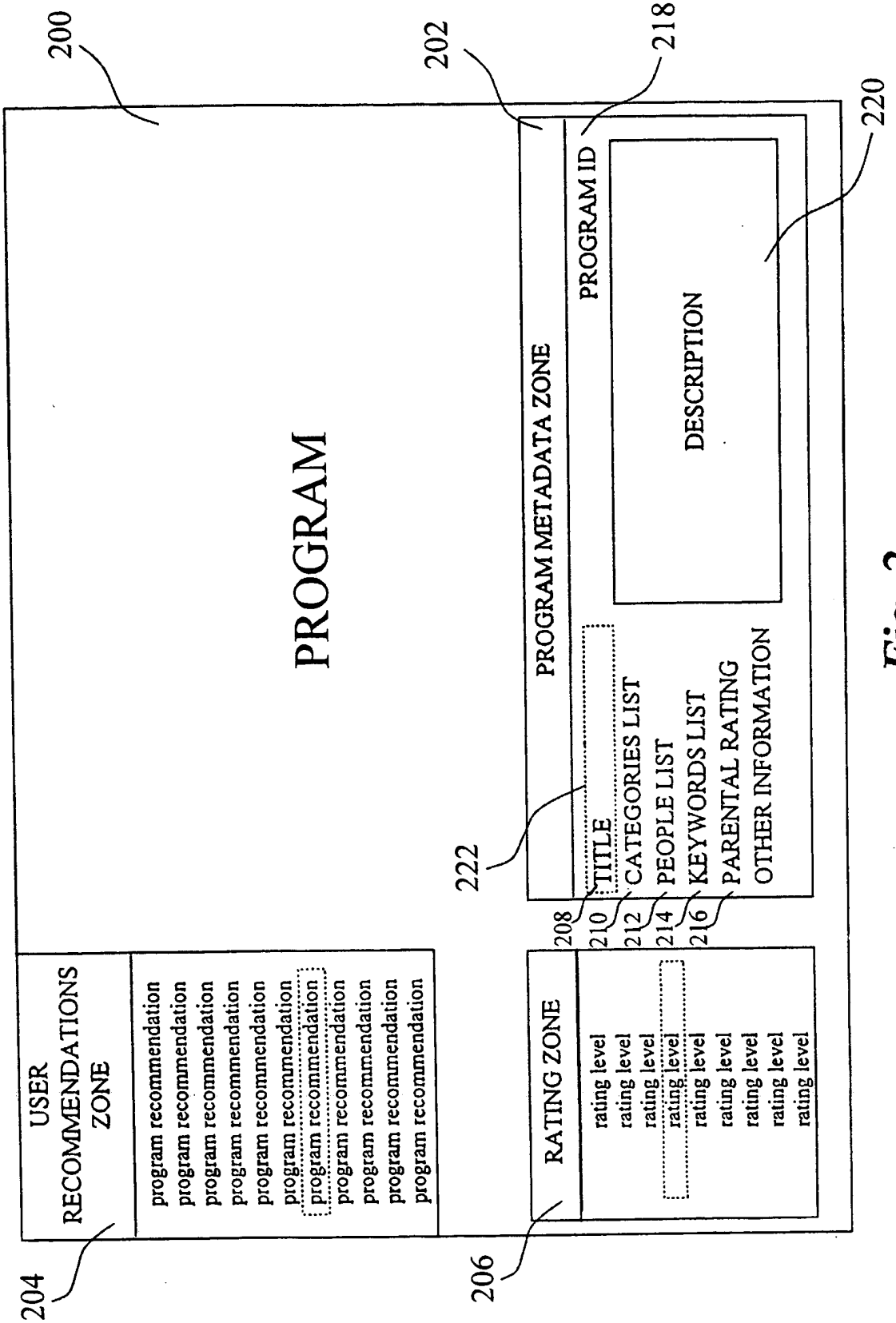

200
PROGRAM
202
218
220
PROGRAM METADATA ZONE
PROGRAM ID
DESCRIPTION
222
TITLE
CATEGORIES LIST
PEOPLE LIST
KEYWORDS LIST
PARENTAL RATING
OTHER INFORMATION
208
210
212
214
216
USER RECOMMENDATIONS ZONE
program recommendation
RATING ZONE
rating level
204
206

Fig. 2

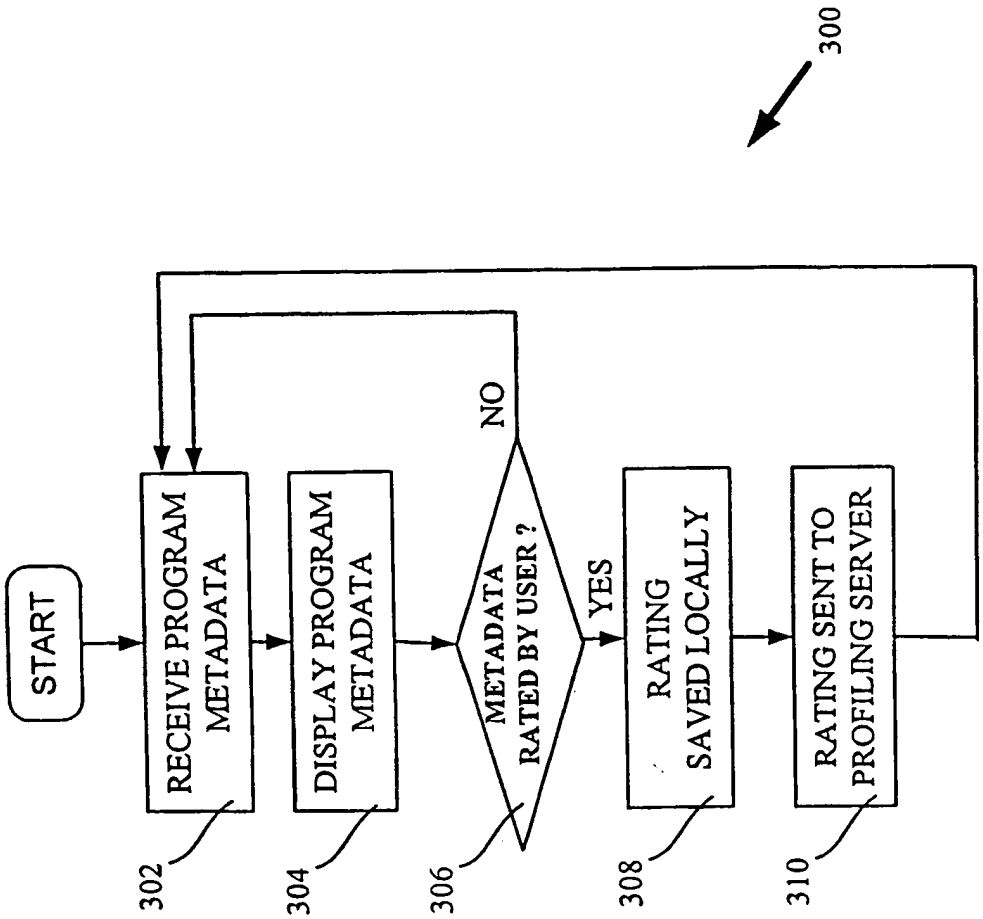

300
START
RECEIVE PROGRAM METADATA
DISPLAY PROGRAM METADATA
METADATA RATED BY USER ?
NO
YES
RATING SAVED LOCALLY
RATING SENT TO PROFILING SERVER
302
304
306
308
310

Fig. 3

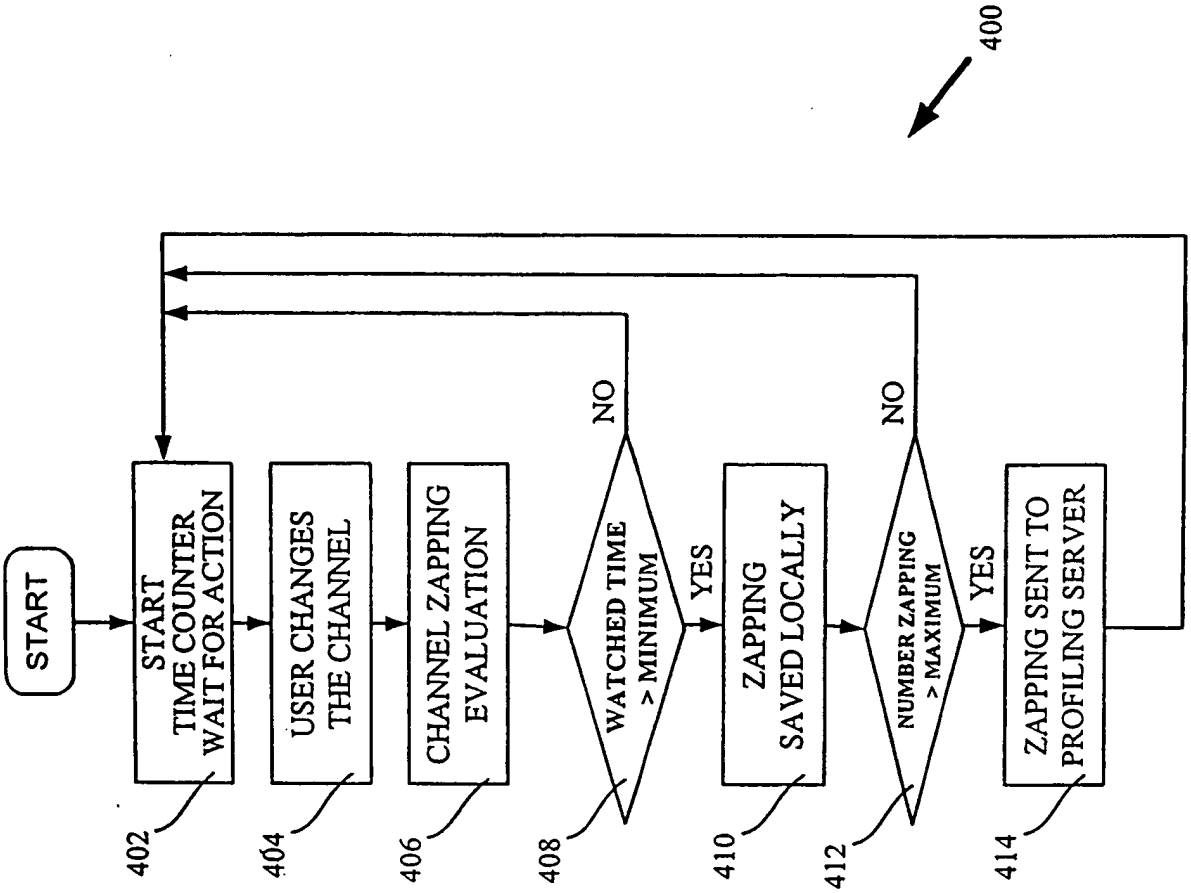

400
START
START TIME COUNTER WAIT FOR ACTION
USER CHANGES THE CHANNEL
CHANNEL ZAPPING EVALUATION
WATCHED TIME > MINIMUM
NO
YES
ZAPPING SAVED LOCALLY
NUMBER ZAPPING > MAXIMUM
NO
YES
ZAPPING SENT TO PROFILING SERVER
402
404
406
408
410
412
414

Fig. 4

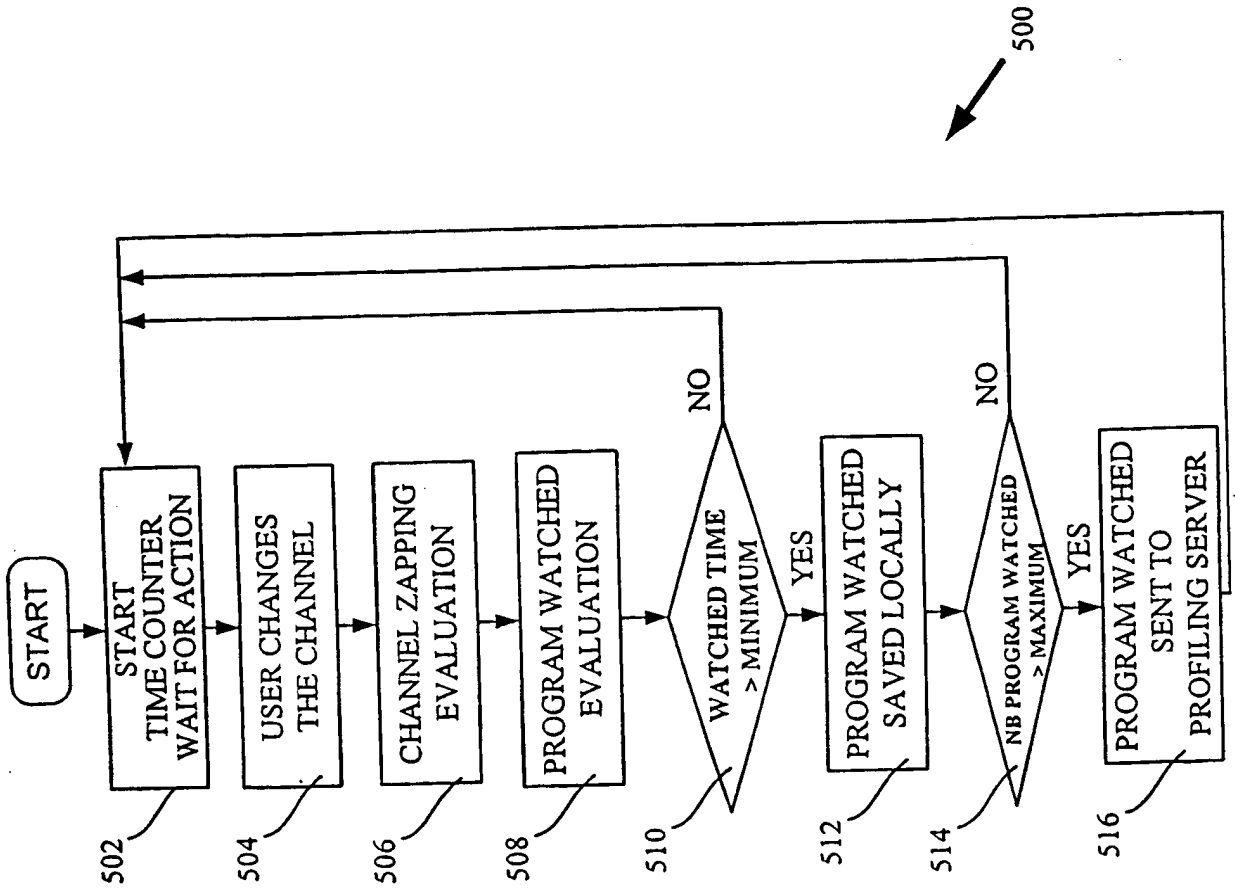

500
START
START TIME COUNTER WAIT FOR ACTION
USER CHANGES THE CHANNEL
CHANNEL ZAPPING EVALUATION
PROGRAM WATCHED EVALUATION
WATCHED TIME > MINIMUM
NO
YES
PROGRAM WATCHED SAVED LOCALLY
NB PROGRAM WATCHED > MAXIMUM
NO
YES
PROGRAM WATCHED SENT TO PROFILING SERVER
502
504
506
508
510
512
514
516

Fig. 5

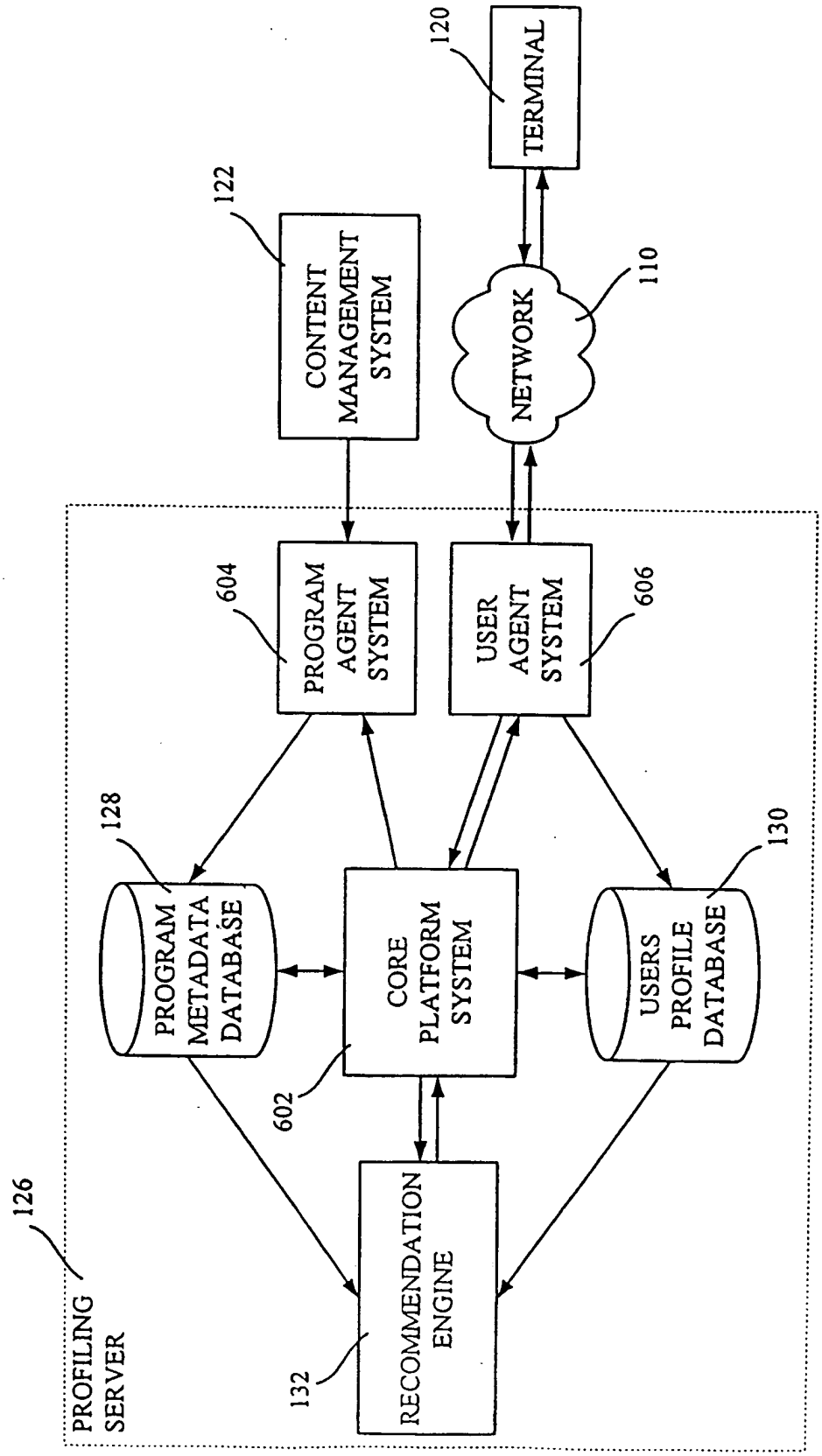

600
120
TERMINAL
122
CONTENT MANAGEMENT SYSTEM
110
NETWORK
604
PROGRAM AGENT SYSTEM
606
USER AGENT SYSTEM
128
PROGRAM METADATA DATABASE
CORE PLATFORM SYSTEM
130
USERS PROFILE DATABASE
602
126
PROFILING SERVER
RECOMMENDATION ENGINE
132

Fig. 6

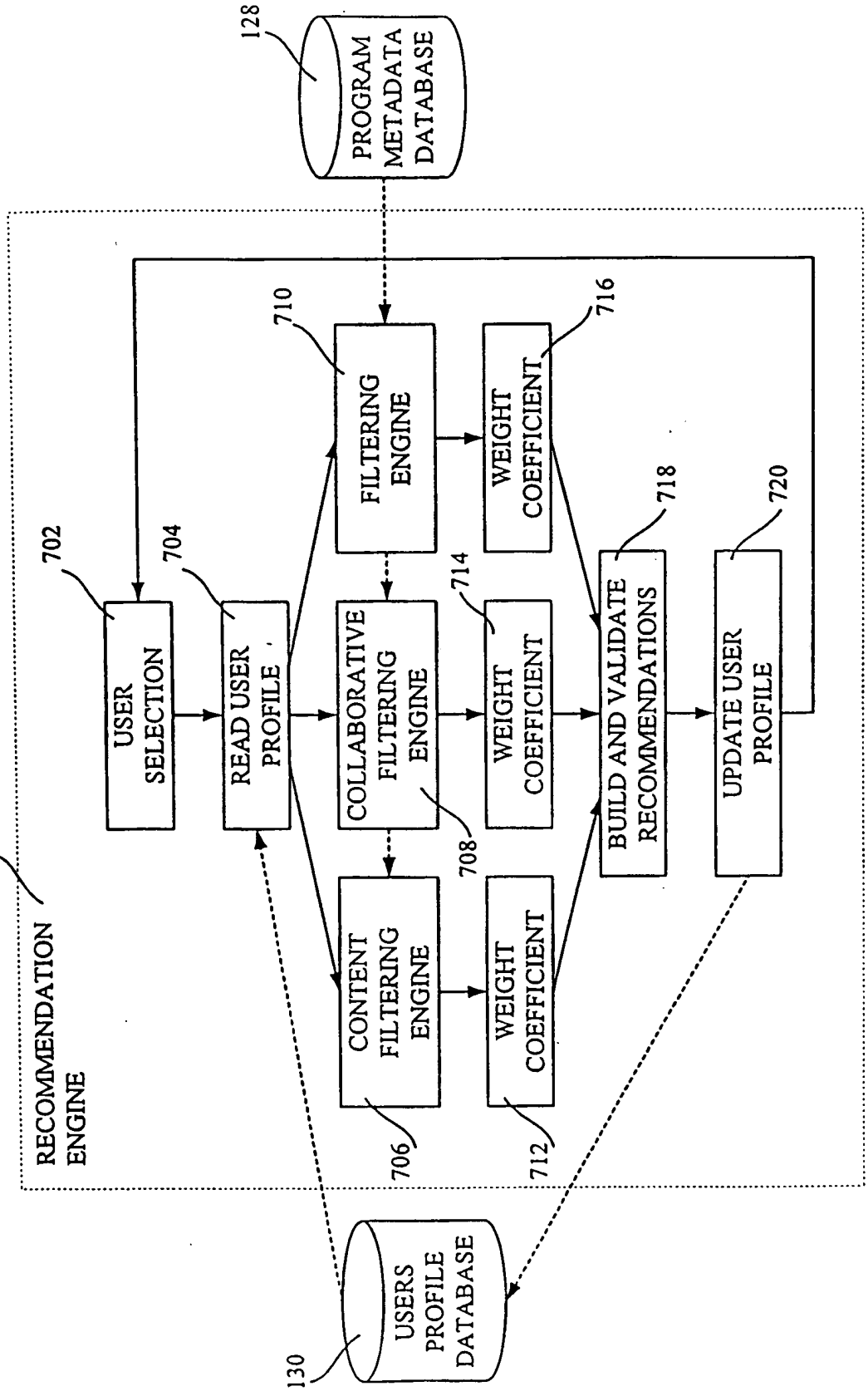

128
PROGRAM METADATA DATABASE
710
FILTERING ENGINE
716
WEIGHT COEFFICIENT
702
USER SELECTION
704
READ USER PROFILE
714
COLLABORATIVE FILTERING ENGINE
WEIGHT COEFFICIENT
718
BUILD AND VALIDATE RECOMMENDATIONS
720
UPDATE USER PROFILE
132
RECOMMENDATION ENGINE
708
CONTENT FILTERING ENGINE
WEIGHT COEFFICIENT
706
712
130
USERS PROFILE DATABASE

Fig. 7

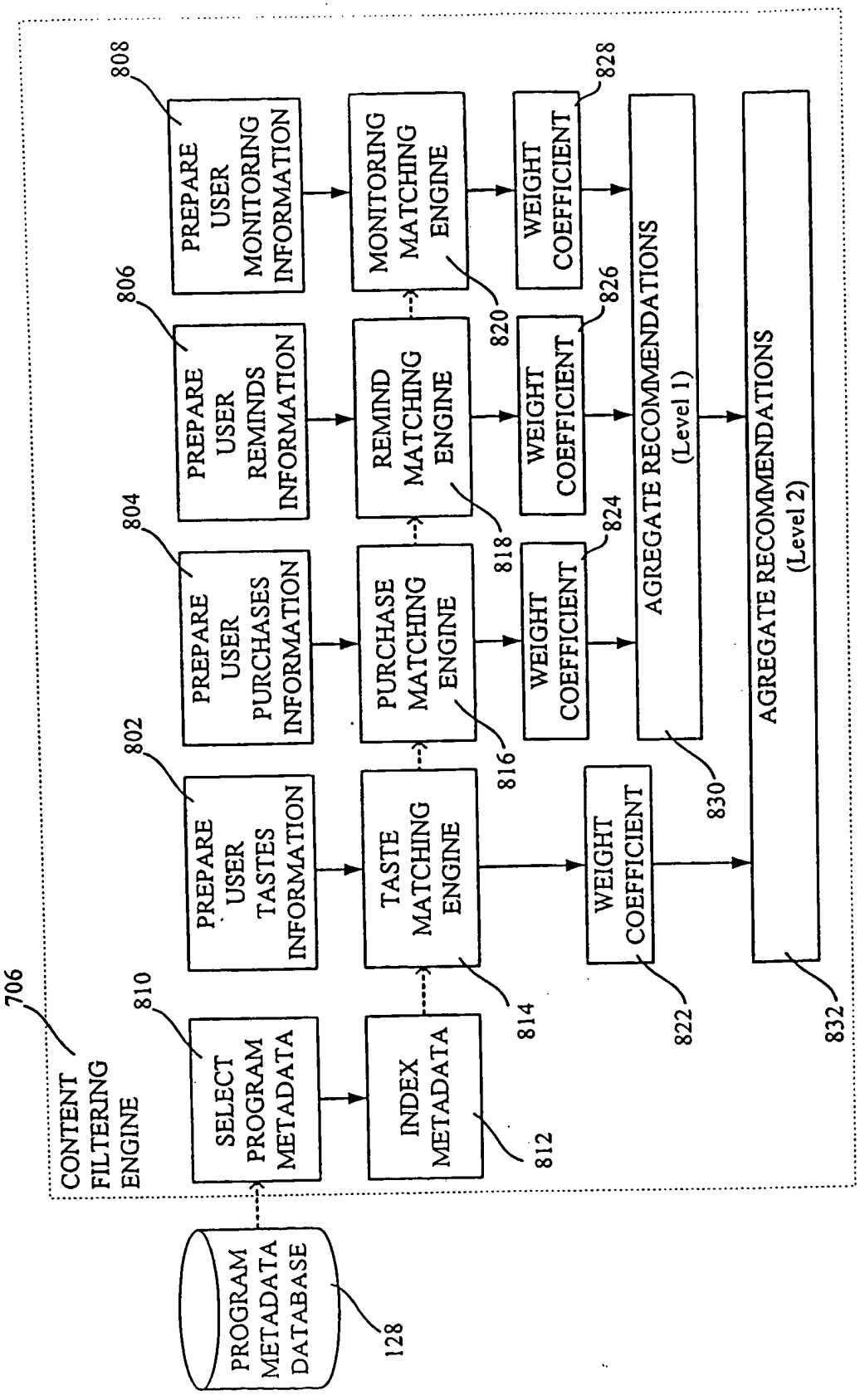

CONTENT FILTERING ENGINE
706
PROGRAM METADATA DATABASE
128
SELECT PROGRAM METADATA
810
INDEX METADATA
812
PREPARE USER TASTES INFORMATION
802
TASTE MATCHING ENGINE
814
PREPARE USER PURCHASES INFORMATION
804
PURCHASE MATCHING ENGINE
816
PREPARE USER REMINDS INFORMATION
806
REMIND MATCHING ENGINE
818
PREPARE USER MONITORING INFORMATION
808
MONITORING MATCHING ENGINE
820
WEIGHT COEFFICIENT
822
WEIGHT COEFFICIENT
824
WEIGHT COEFFICIENT
826
WEIGHT COEFFICIENT
828
AGREGATE RECOMMENDATIONS (Level 1)
830
AGREGATE RECOMMENDATIONS (Level 2)
832

Fig. 8

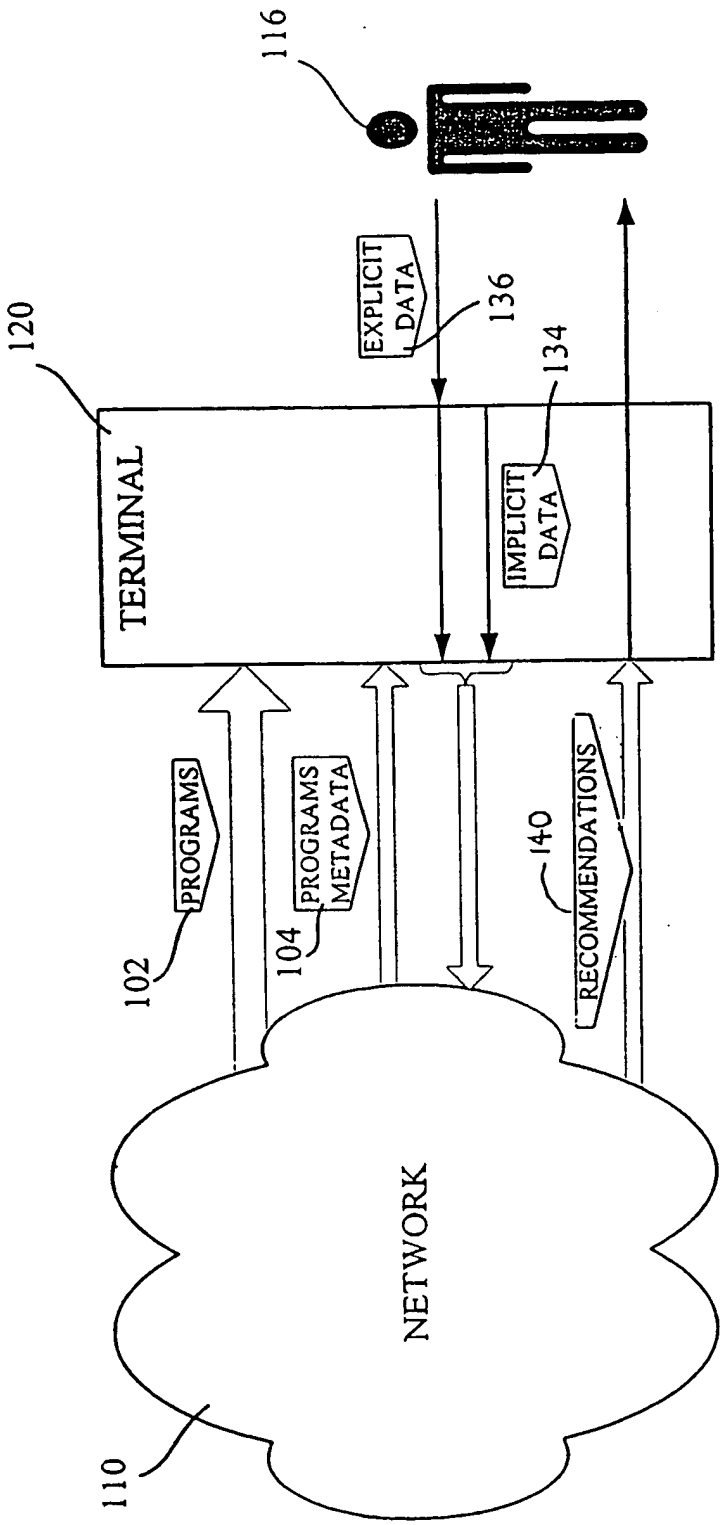

116
EXPLICIT DATA
136
134
120
TERMINAL
IMPLICIT DATA
PROGRAMS
PROGRAMS METADATA
RECOMMENDATIONS
140
102
104
NETWORK
110

Fig. 9

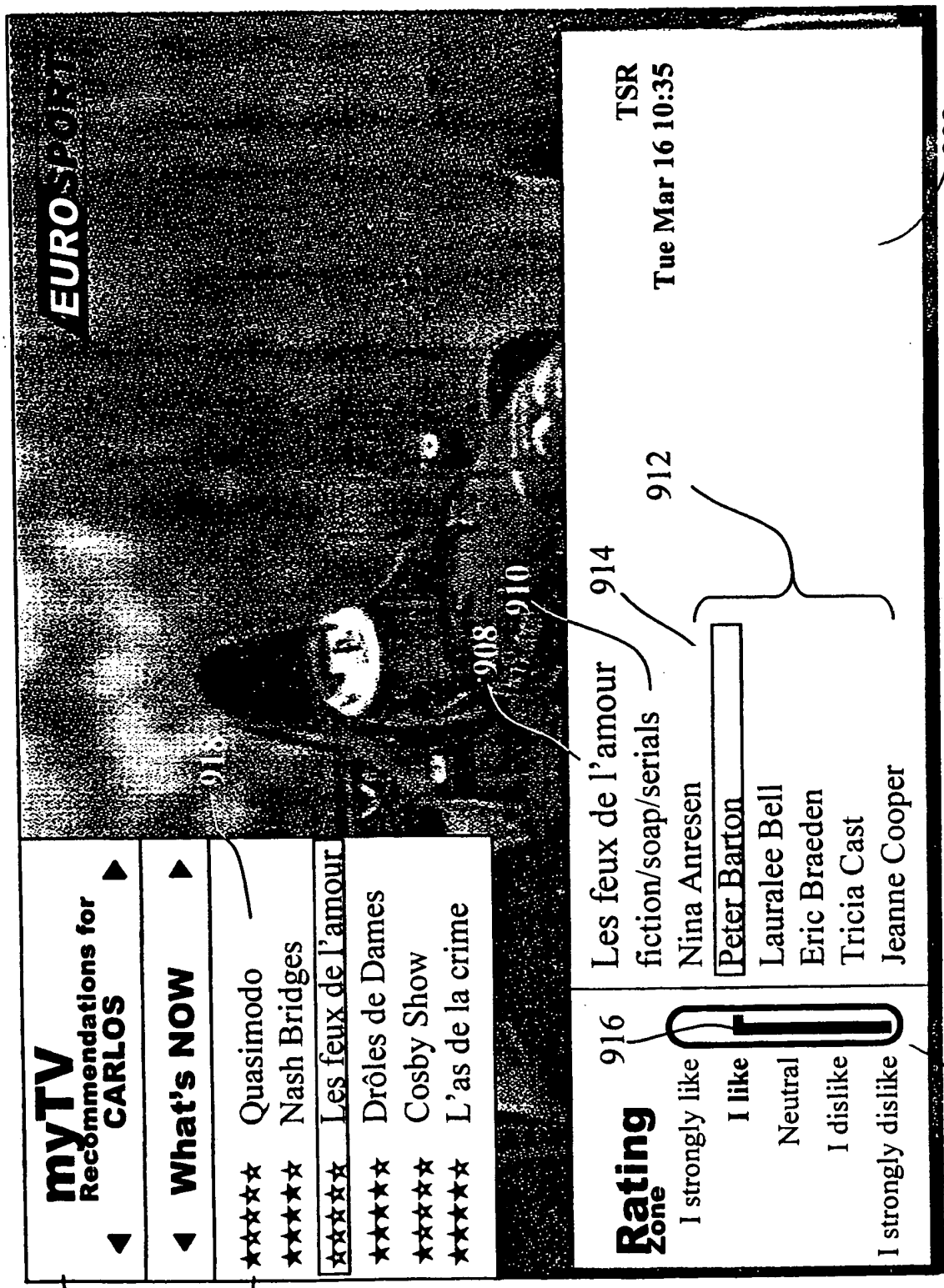

myTV
Recommendations for CARLOS
What's NOW
★★★★★ Quasimodo
★★★★★ Nash Bridges
★★★★★ Les feux de l'amour
★★★★★ Drôles de Dames
★★★★★ Cosby Show
★★★★★ L'as de la crime
EURO
Les feux de l'amour
fiction/soap/serials
Nina Anresen
Peter Barton
Lauralee Bell
Eric Braeden
Tricia Cast
Jeanne Cooper
TSR
Tue Mar 16 10:35
Rating Zone
I strongly like
I like
Neutral
I dislike
I strongly dislike
906
920
916
914
912
910
908
904
902

Fig. 10

INTERNATIONAL SEARCH REPORT

Inte ional Application No
PCT/IB 00/01131

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04N7/173 H04N7/16

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 H04N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, WPI Data, PAJ

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5 758 257 A (HERZ FREDERICK ET AL) 26 May 1998 (1998-05-26)	1,2,7-12
Y	column 11, line 45 -column 12, line 6 column 13, line 29 - line 54 column 19, line 6 - line 13 column 34, line 12 - line 23 column 40, line 11 - line 20 ---	3-6
	-/--	

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
13 October 2000	20/10/2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31–70) 340–2040, Tx. 31 651 epo nl, Fax: (+31–70) 340–3016	Sindic, G

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Int. ional Application No
PCT/IB 00/01131

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
X	WITTIG H ET AL: "INTELLIGENT MEDIA AGENTS IN INTERACTIVE TELEVISION SYSTEMS" PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON MULTIMEDIA COMPUTING AND SYSTEMS,US,LOS ALAMITOS, CA, 15 May 1995 (1995-05-15), pages 182-189, XP000603484 page 183, left-hand column, paragraph 2 page 184, right-hand column, paragraph 2 page 185, right-hand column, paragraph 3 -page 187, right-hand column, paragraph 4	1,4-6,9
Y	NYGREN K ET AL: "An Agent System For Media On Demand Services" PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON THE PRACTICAL APPLICATION OF INTELLIGENT AGENTS AND MULTI-AGENT TECHNOLOGY, 22 April 1996 (1996-04-22), pages 437-453, XP002086093 page 440, paragraph 1 - paragraph 2 page 441, paragraph 6 page 443, paragraph 3 page 445, paragraph 4 - paragraph 5 page 446, paragraph 3 page 447, paragraph 3 page 449, paragraph 2 - paragraph 4	3-6
A	US 5 798 785 A (BONNER ALFRED E ET AL) 25 August 1998 (1998-08-25) column 29, line 26 -column 30, line 39 column 32, line 54 - line 62	1,9

INTERNATIONAL SEARCH REPORT

Information on patent family members

Int. ional Application No
PCT/IB 00/01131

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5758257 A	26-05-1998	AU 703247 B	25-03-1999
		AU 4410396 A	19-06-1996
		CA 2207868 A	06-06-1996
		EP 0796538 A	24-09-1997
		US 6020883 A	01-02-2000
		WO 9617467 A	06-06-1996
		US 5734720 A	31-03-1998
		US 5754938 A	19-05-1998
		US 5754939 A	19-05-1998
		US 5835087 A	10-11-1998
		US 6088722 A	11-07-2000
		US 6029195 A	22-02-2000
US 5798785 A	25-08-1998	AU 1333795 A	19-06-1995
		AU 693148 B	25-06-1998
		AU 1430695 A	19-06-1995
		BR 9408211 A	26-08-1997
		CA 2177153 A	08-06-1995
		CA 2177154 A	08-06-1995
		EP 0732027 A	18-09-1996
		EP 0732031 A	18-09-1996
		EP 0963116 A	08-12-1999
		JP 9506225 T	17-06-1997
		JP 9510327 T	14-10-1997
		NZ 278185 A	27-04-1998
		WO 9515649 A	08-06-1995
		WO 9515658 A	08-06-1995
		AT 177277 T	15-03-1999
		AT 176840 T	15-03-1999
		AT 192005 T	15-05-2000
		AT 190180 T	15-03-2000
		AT 183352 T	15-08-1999
		AT 176841 T	15-03-1999
		AU 715683 B	10-02-2000
		AU 4440797 A	29-01-1998
		AU 712157 B	28-10-1999
		AU 4532597 A	05-02-1998
		AU 693775 B	09-07-1998
		AU 5732994 A	04-07-1994
		AU 692427 B	11-06-1998
		AU 5733094 A	04-07-1994
		AU 691479 B	21-05-1998
		AU 5733194 A	04-07-1994
		AU 692428 B	11-06-1998
		AU 5733294 A	04-07-1994
		AU 5736394 A	04-07-1994
		AU 5845894 A	22-06-1994
		AU 5869894 A	04-07-1994
		AU 716184 B	24-02-2000
		AU 6066798 A	04-06-1998
		AU 716182 B	24-02-2000
		AU 6066898 A	04-06-1998
		BR 9307619 A	15-06-1999
		BR 9307620 A	10-08-1999
		BR 9307621 A	15-06-1999
		BR 9307622 A	15-06-1999
		BR 9307623 A	16-05-2000
		BR 9307624 A	15-06-1999

INTERNATIONAL SEARCH REPORT

Information on patent family members

Int .tional Application No
PCT/IB 00/01131

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5798785 A		BR 9307625 A	31-08-1999
		CA 2151456 A	23-06-1994
		CA 2151457 A	23-06-1994

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.